

**DESCRIPTION OF X2000 COMPONENTS
AVAILABLE FOR USE IN INSTRUMENT PROPOSALS**

June 1999

DESCRIPTION OF X2000 COMPONENTS AVAILABLE FOR USE IN INSTRUMENT PROPOSALS

Table 1 provides a preliminary description of X2000 developed items that may be useful for Science Instrument builders. This description is intended to provide an introduction to the current preliminary designs for these components. These designs may change over time.

None of these parts presently exist. Specifications are X2000 current best estimates usable for planning purposes; proposers should include appropriate margins in their designs.

Table 1: X2000 Parts Under Development and Available for Science Instrument Use

<p><u>1) Generic Microcontroller</u> Smart interface between load and system data buses</p> <ul style="list-style-type: none"> • Mass 0.6 kg (does not include tbd shielding mass for non-volatile memory); Cost \$200K (planning estimate) • Basic Features <ul style="list-style-type: none"> • 32-bit microcontroller with 10 MIPS • 128KB PROM, 2MB SRAM • Dual isolated 1394 and I²C interfaces • 2 S/W-development UARTs (not available for general use) • 2KB non-volatile memory • Externally Accessible Buses <ul style="list-style-type: none"> • Single 3.3V @ 30W power output • PCI bus • Processor Local Bus • 2 additional user-defined UART ports and 2 I²C subnet ports • 1 MRad hard (except for non-volatile memory) • Compact PCI package • Earliest availability: Prototype 5/00, EM 10/00, Flight 4/01 • Contact: JPL- Jeff Bowers; TRW 	<p><u>3) Non-volatile Memory (NVM) Slice</u></p> <ul style="list-style-type: none"> • Mass 0.25 kg: Cost \$75K (planning estimate) • Power 3.0 W (active) 0.7 W (inactive) • Capacity: 256 MB, memory mapped; 100,000 write cycles • Busses: 33MHz PCI, JTAG standard test interface • Features: <ul style="list-style-type: none"> • error detection and correction code • 1 MB non-volatile memory; unlimited write cycles • Compact PCI package • Europa shielding mass required: TBD (>4kg). Radiation damage is less with power not applied to unit, thus power cycling may be required to reduce overall radiation damage • Earliest availability: EM 6/00, Flight 1/01 • Contact: JPL-Karl Strauss; SEAKR 																
<p><u>2) Power Converter</u></p> <ul style="list-style-type: none"> • Mass 0.13 kg: Cost \$75K (planning estimate) • Power Converter Unit • Input Bus Range 18V - 36V • Single Output 3.3V (30W); other voltage and power levels may be available at a later date • Regulation $\pm 3\%$ (EOL) • Ripple 50mVp-p • Size 1.5" x 3.0" x 0.4" • Efficiency 65% @ 5W, 85% @ 30W (full load) • Output over voltage and short circuit protection • 1 MRad hard • Class S • Earliest availability: EM 10/00, Flight 4/01 • Contact: JPL-Phil Brisendine; Lockheed Martin-Communications and Power Center, Shey Sabripour 	<p>None of these parts presently exist. Specifications are X2000 current best estimates usable for planning purposes; proposers should include appropriate margins in their designs.</p> <table> <tr> <td>EM =</td><td>Engineering Model</td></tr> <tr> <td>EOL=</td><td>End of Life</td></tr> <tr> <td>JTAG =</td><td>Joint Test Action Group</td></tr> <tr> <td>KB =</td><td>kilobytes</td></tr> <tr> <td>MIPS =</td><td>Million Instructions per Second</td></tr> <tr> <td>PROM =</td><td>Programmable Read Only Memory</td></tr> <tr> <td>SRAM =</td><td>Static Random Access Memory</td></tr> <tr> <td>UART =</td><td>Universal Asynchronous Receiver Transmitter</td></tr> </table>	EM =	Engineering Model	EOL=	End of Life	JTAG =	Joint Test Action Group	KB =	kilobytes	MIPS =	Million Instructions per Second	PROM =	Programmable Read Only Memory	SRAM =	Static Random Access Memory	UART =	Universal Asynchronous Receiver Transmitter
EM =	Engineering Model																
EOL=	End of Life																
JTAG =	Joint Test Action Group																
KB =	kilobytes																
MIPS =	Million Instructions per Second																
PROM =	Programmable Read Only Memory																
SRAM =	Static Random Access Memory																
UART =	Universal Asynchronous Receiver Transmitter																